

方案设计文档

1、根据sdk指引，使用openapi接口

1.请求 openapi 接口，操作资源

```
func main() {
    token := token.BotToken(conf.AppID, conf.Token)
    api := botgo.NewOpenAPI(token).WithTimeout(3 * time.Second)
    ctx := context.Background()

    ws, err := api.WS(ctx, nil, "")
    log.Printf("%+v, err:%v", ws, err)

    me, err := api.Me(ctx, nil, "")
    log.Printf("%+v, err:%v", me, err)
}
```

```
func main() {
    token := token.BotToken(config.AppID, config.Token)
    //获取沙箱的api
    //api = botgo.NewOpenAPI(token).WithTimeout(3 * time.Second)
    api = botgo.NewSandboxOpenAPI(token).WithTimeout(3 * time.Second)
    ctx = context.Background()
    ws, err := api.WS(ctx, params: nil, body: "")
    if err != nil {
        log.Fatalln(v...: "websocket错误, err = ", err)
        os.Exit(code: 1)
    }
}
```

2、监听websocket AtMessage事件

```
var atMessage event.ATMessageEventHandler = atMessageEventHandler
/* var atMessage event.ATMessageEventHandler = func(event *dto.WSPayload, data
fmt.Println("event:", event, "\ndata:", data)
punch(data.ChannelID, data.ID)
return nil
}*/
intent := event.RegisterHandlers(atMessage) // 注册socket消息处理
botgo.NewSessionManager().Start(ws, token, &intent) // 启动socket监听
}
```

3、根据at 机器人的消息Content，处理不同的功能

1、打卡功能

用map记录每个user的每天是否打卡，如果当天未打卡，打卡成功，否则提示“请勿重复打卡”

2、查询打卡信息

获取当前用户的打卡情况并回传

3、问候功能

当收到"你好"问候时，回传"您好"